

DESENVOLVIMENTO DE UMA IDE WEB COM USO DO JAVAFX

Eduardo Cardoso¹

¹ Instituto Federal Catarinense - Bacharelado em Ciência da Computação
edu.cardoso516@gmail.com

Abstract. *During object-oriented programming, many techniques and patterns have been studied that show how using object-oriented programming makes work agile according to what has already been produced, as well as reducing the amount of errors and rework with use of UML. Although theoretical knowledge is important, these techniques must be implemented in practice to be effective, so a development project was proposed where these concepts should be implemented. This project involves the development of an aid tool for editing and creating web pages.*

Keywords: *Object Oriented Programming; Web development; GitHub;*

Resumo. *Durante a matéria de Programação Orientado a objetos, foram estudadas muitas técnicas e padrões que mostram como o uso de programação voltada a objetos torna o trabalho ágil de acordo com o que já foi produzido , assim como diminuir a quantidade de erros e de retrabalho com a utilização de UML. Por mais que o conhecimento teórico seja importante, essas técnicas devem ser implementadas na prática para que sejam eficazes, por isso foi proposto um projeto de desenvolvimento onde deveriam ser implementados esses conceitos. Esse projeto consiste no desenvolvimento de uma ferramenta de auxílio para edição e criação de páginas web.*

Palavras-chave: *Programação Orientada a Objetos; Desenvolvimento web; GitHub;*

1. Introdução

Durante a matéria de Programação Orientado a objetos, foram estudadas muitas técnicas e padrões que mostram como o uso de programação voltada a objetos torna o trabalho ágil de acordo com o que já foi produzido , assim como diminuir a quantidade de erros e de retrabalho com a utilização de UML. Por mais que o conhecimento teórico seja importante, essas técnicas devem ser implementadas na prática para que sejam eficazes, por isso foi proposto um projeto de desenvolvimento onde deveriam ser implementados esses conceitos. Esse projeto consiste no desenvolvimento de uma

ferramenta de auxílio para edição e criação de páginas web.

O projeto consiste no desenvolvimento de uma ferramenta com três campos editáveis que tem por finalidade a edição HTML, CSS, JavaScript, a IDE utilizada para o desenvolvimento base foi o NetBeans IDE versão 8.2, SceneBuilder para a parte de geração e estruturação da interface Gráfica e StarUML para criação do diagrama de classes . A descrição do projeto foi realizada com o PM Canvas, o gerenciamento de desenvolvimento foi realizado com Scrum e Kanban com o uso do Trello como ferramenta, enquanto o gerenciamento de versões foi feito com o GitHub.

2. Fundamentação Teórica

Abaixo são descritas as linguagens e ferramentas que foram utilizadas na construção da ferramenta.

JAVA Java é uma linguagem de programação e plataforma computacional lançada pela primeira vez pela Sun Microsystems em 1995.

HTML (*HyperText Markup Language*), É uma linguagem de marcação utilizada na construção de páginas na Web. Documentos HTML podem ser interpretados por navegadores, como são arquivos de texto simples, podem ser criados e editados em qualquer editor de textos comum, como o Bloco de Notas.

CSS (*Cascading Style Sheets*), é uma folha de estilos em cascatas, que possibilita a estilização dos elementos escritos no HTML. Com ele foi possível alterar cor do texto, colocar imagem de fundo, fonte e espaçamento entre linhas e parágrafos, alinhamento dos elementos, ajustar imagens conforme as telas e muitas outras coisas.

JavaScript, é utilizado para controlar o HTML e o CSS para manipular comportamentos na página, como no caso de um menu, quando passa o mouse, por exemplo, e ele muda de cor ou acrescenta um elemento com cor para estilizar.

Com auxílio da ferramenta NetBeans IDE versão 8.2, foi feita a edição do código-fonte das páginas.

O Github foi utilizado para o Armazenamento dos arquivos.

3. Desenvolvimento

Logo abaixo vemos as disposição das classes

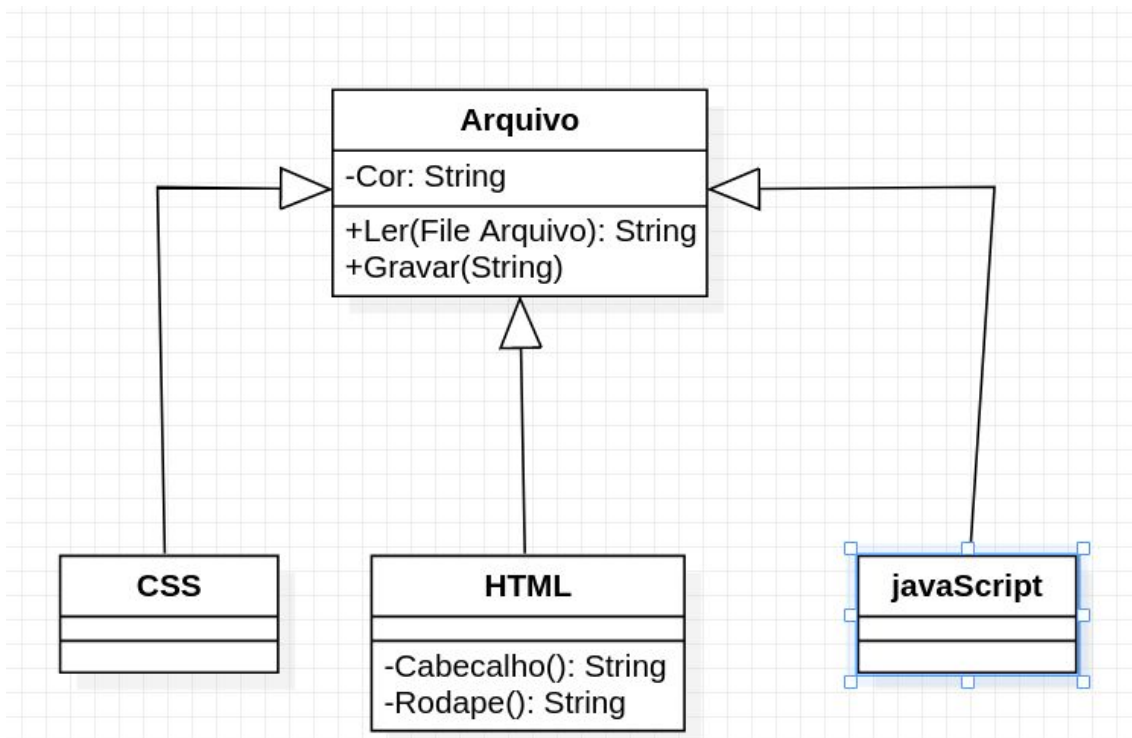


Figura 1 – Diagrama de Classe(Fonte:Autor)

```

6 package Arquivos;
7
8 import java.io.BufferedReader;
9 import java.io.File;
10 import java.io.FileNotFoundException;
11 import java.io.FileReader;
12 import java.io.IOException;
13 import java.util.logging.Level;
14 import java.util.logging.Logger;
15
16 /**
17  *
18  * @author eduardo
19  */
20 public abstract class Arquivo {
21     private String cod;
22
23     public String getCod() {
24         return cod;
25     }
26
27     public void setCod(String cod) {
28         this.cod = cod;
29     }
30
31     public abstract String ler(File Arquivo);
32     public abstract void gravar(String cod) throws IOException;
33
  
```

Figura 2 – Classe Arquivo (Fonte:Autor)

Na Figura 2, vemos a classe arquivo responsável pela parte de receber e setar código, tendo dois métodos abstratos(Esses métodos obrigam as classes que herdam a mesma, tem que obrigatoriamente implementá-los)

```

16 private static String cabecalho() {
17     StringBuilder html = new StringBuilder();
18     html.append("<!DOCTYPE html>\n");
19     html.append("<html>\n");
20     html.append("<head><title>Projeto</title>\n");
21     html.append("<meta charset='UTF-8'>\n");
22     //para fazer a utilização de aspas no java "/"eu estou entre aspas""
23     html.append("<link rel='stylesheet' type='text/css' href='style0.css'>");
24     html.append("\n<script type='text/javascript' src='Script0.js'></script>");
25     html.append("</head>\n");
26     html.append("<body>\n");
27     return html.toString();
28 }
29
30 private static String rodape() {
31     StringBuilder html = new StringBuilder();
32     html.append("</body>\n");
33     html.append("</html>\n");
34     return html.toString();
35 }

```

Figura 3 – Classe HTML (Fonte:Autor)

Na Figura 3, temos a implementação de dois métodos extremamente importantes, visto que eles fazem a parte de requisição/localização dos arquivos css e javascript, sem eles nossa IDE não vai poder apresentar sua principal função que é a edição de código direto no corpo de uma página.

```

150 @Override
151 public void initialize(URL location, ResourceBundle resources) {
152     Alert dialogInfo = new Alert(Alert.AlertType.INFORMATION);
153     dialogInfo.setTitle("Bem-Vindo IDEWeb");
154     dialogInfo.setHeaderText("Localize ou crie uma pagina html inicial");
155     dialogInfo.showAndWait();
156     FileChooser importar = new FileChooser();
157     importar.setTitle("Importar");
158     Window Stage = null;
159     WebEngine navegador = Navegador.getEngine();
160     uri = "file://" + importar.showOpenDialog(Stage);
161     navegador.load(uri);
162     navegador.setJavaScriptEnabled(true);
163 }
164

```

Figura 4 – graficoController (Fonte:Autor)

Na Figura 4, temos a implementação dos métodos responsáveis por interação com usuário dentro da interface gráfica, aqui temos algumas partes que dão vida a nossa ferramenta, como a WebEngine(Biblioteca disponível e incluída dentro do JDK, tem como função a execução e criação de programas voltadas a Web, muitas vezes fazendo o papel de “Navegador Web”).

Ainda na Figura 4, encontramos o FileChooser(Escolhedor de arquivos incluso dentro do JDK).Tem como função fazer a ligação com a maquina sendo utilizada para busca de arquivos externos.

4. Resultado

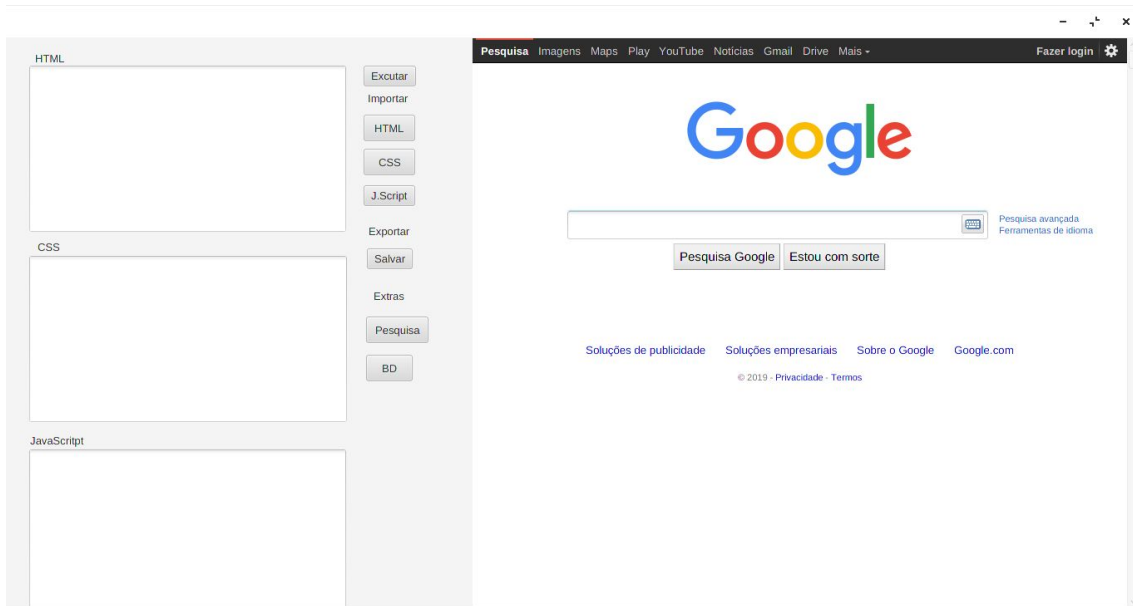


Figura 5 – IDEWeb(Fonte:Autor)

Figura 5, temos finalmente nosso resultado, temos uma ferramenta com três áreas de edição, no seu centro temos sua central de comando, desde execução, a extras como gerenciamento de Banco de dados e pesquisa na internet através do site de busca da Google.

5. Conclusão

Em geral, teve a utilização de técnicas como Engenharia de Software foram de grande valia para o processo de desenvolvimento, fornecendo benefícios consideráveis com o seu uso, e os obstáculos apresentados foi a utilização das bibliotecas java muitas vezes ter que parar grande parte do tempo para leitura para melhor entendimento, de modo que, com o decorrer do projeto, teve certas aquisições de conhecimento como melhor gerenciamento de versão com o git sobre as várias versões do projeto. E obtendo uma ferramenta que tem grandes áreas para melhorar e sendo cada vez mais útil.

Referências

«**When it comes to reproducible science, Git is code for success**». www.natureindex.com. Consultado em 19 de novembro de 2019.

«**About**». GitHub (em inglês). Consultado em 19 de novembro de 2019.

CAETANO, Cristiano (2004). **CVS. Controle de Versões e Desenvolvimento Colaborativo de Software**. Editora Novatec;

«**Ecommerce Video Games Responsive Template**». <<https://www.themezy.com/free-website-templates/11-ecommerce-video-games-responsive-template>>. Consultado em 26 de novembro de 2019.

«**JavaFX WebView and JavaFX WebView and WebEngine Tutorial I**». <<https://o7planning.org/en/11151/javafx-webview-and-webengine-tutorial>>. Consultado em 20 de novembro de 2019.

«**Usando controles de interface de usuário JavaFX**». <https://docs.oracle.com/javafx/2/ui_controls/file-chooser.htm>. Consultado em 20 de novembro de 2019;